Image Conductor: Precision Control for Interactive Video Synthesis

Yaowei Li[1,3], Xintao Wang[2], Zhaoyang Zhang [2‡], Zhouxia Wang[4], Ziyang Yuan[5], Liangbin Xie[6,7], Ying Shan[2], Yuexian Zou[1,3*]

[1]School of Electronic and Computer Engineering, Peking University, Shenzhen, China
[2]ARC Lab, Tencent PCG, Shenzhen, China
[3] Guangdong Provincial Key Laboratory of Ultra High Definition Immersive Media Technology, Peking University Shenzhen Graduate School, Shenzhen, China
[4] Nanyang Technological University, Singapore
[5] Shenzhen International Graduate School, Tsinghua University, Shenzhen, China
[6] University of Macau, Macao SAR
[7]Shenzhen Institute of Advanced Technology, Shenzhen, China

Abstract

Filmmaking and animation production often require sophisticated techniques for coordinating camera transitions and object movements, typically involving labor-intensive real-world capturing. Despite advancements in generative AI for video creation, achieving precise control over motion for interactive video asset generation remains challenging. To this end, we propose Image Conductor, a method for precise control of camera transitions and object movements to generate video assets from a single image. A well-cultivated training strategy is proposed to separate distinct camera and object motion by camera LoRA weights and object LoRA weights. To further eliminate motion ambiguity from ill-posed trajectories, we introduce a camera-free guidance technique during inference process, enhancing object movements while eliminating camera transitions. Additionally, we develop a trajectory-oriented video motion data curation pipeline for training. Quantitative and qualitative experiments demonstrate our method's precision and fine-grained control in generating motion-controllable videos from images, advancing the practical application of interactive video synthesis.

Project Page —
https://liyaowei-stu.github.io/project/ImageConductor/

Introduction

Filmmaking and animation production are essential forms of visual art. During the creative process of video media, professional directors often require advanced cinematography techniques to meticulously plan and coordinate camera transitions and object movements, ensuring storyline coherence and refined visual effects. To achieve precise creative expression, the current workflow for video media orchestration and production heavily relies on real-world capturing and 3D scan modeling, which are labor-intensive and costly.

Recent work (Ho et al. 2022; Blattmann et al. 2023b; Girdhar et al. 2023; Xing et al. 2023; Chen et al. 2023; Blattmann et al. 2023a; Bar-Tal et al. 2024; Brooks et al. 2024) explores an AIGC-based filmmaking pipeline that leverages the powerful generative capabilities of diffusion models to generate video clip assets. Despite these advancements, generating dynamic video assets allowing creators precisely express their ideas remains unusable, for: (1) Lacking of efficient generating control interface. (2) Lacking of fine-grained and accurate control over camera transitions and object movements.

Although several works have attempted to introduce motion control signals to guide the video generation process (Yin et al. 2023; Wang et al. 2023, 2024; Wu et al. 2024), none of the existing methods support accurate and fine-grained control over both camera transitions and object movements (see Fig. 1 and Fig. 4).

In fact, data available on the internet often mixes both camera transitions and object movements, leading to ambiguities between the two types of motion. Although MotionCtrl (Wang et al. 2023) uses a data-driven approach to decouple camera transitions from object motion, it still lacks precision and effectiveness. Camera parameters are neither intuitive nor straightforward to obtain for cinematographic variations. For object movements, MotionCtrl uses ParticleSfM (Zhao et al. 2022), a motion segmentation network based on optical flow estimation, which introduces significant errors. Additionally, ground truth videos annotated based on motion segmentation networks still contain camera transitions, causing generated videos to exhibit unintended cinematographic variations. Decoupling cinematographic variations from object movements through data curation is inherently challenging. Obtaining video data from a fixed camera viewpoint, i.e., videos with only object movements, is difficult. Optical flow-based motion segmentation methods (Teed and Deng 2020; Xu et al. 2022; Zhao et al. 2022; Yin et al. 2023; Wang et al. 2023) struggle to accurately track moving objects without errors and fail to eliminate intrinsic camera transitions in realistic videos. Overall, existing methods are either not fine-grained or not sufficiently accurate and effective.

In this paper, we propose Image Conductor, an interactive method for fine-grained object motion and camera control to generate accurate video assets from a single image. Effective fine-grained motion control requires robust motion representation. Trajectories, being intuitive and user-friendly, allow

*‡Project lead. *Corresponding author.

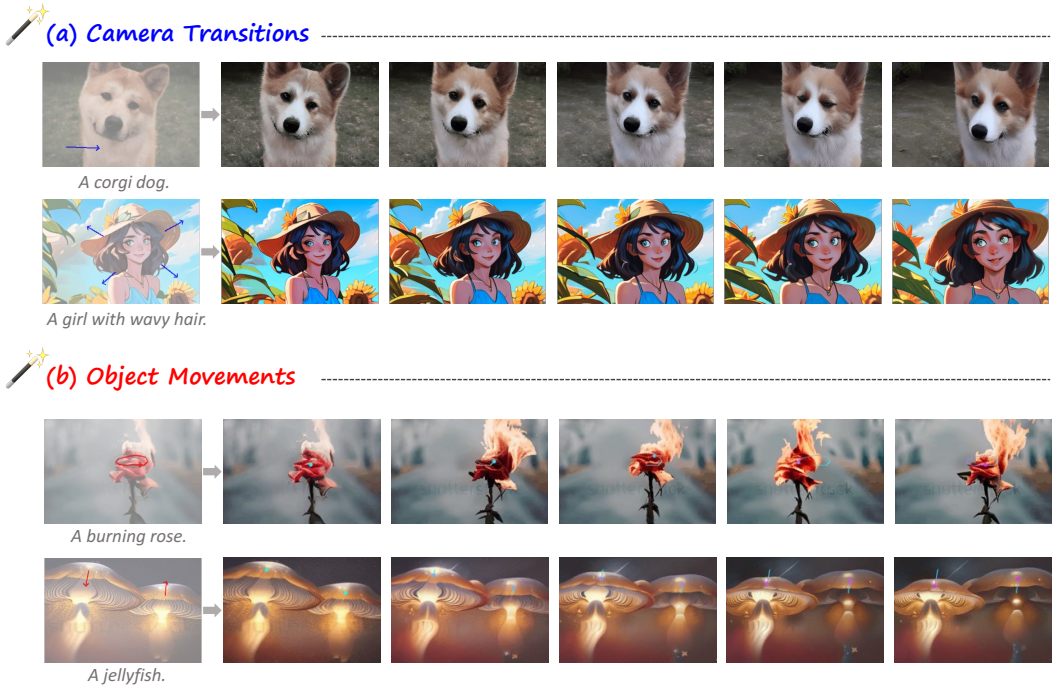

Figure 1: **Orchestrated Results of Image Conductor.** Image Conductor enables fine-grained and accurate image-to-video motion control, including both camera transitions and object movements. Colorful lines denote motion trajectories.

users to control motion in video content by drawing paths. However, a large-scale, high-quality open-source trajectory-based tracking video dataset is currently lacking. To address this, we use CoTracker (Karaev et al. 2023) to annotate existing video data and design a data filtering workflow, resulting in high-quality trajectory-oriented video motion data.

To address the coupling of cinematographic variations and object movements in real-world data, we first train a video ControlNet (Zhang, Rao, and Agrawala 2023) using annotated data to convey motion information to the UNet backbone of the diffusion model. We then propose a collaborative optimization method that applies distinct sets of Low-Rank Adaptation (LoRA) weights (Hu et al. 2021) on the ControlNet to distinguish various types of motion. In addition to the denoising loss commonly used in diffusion models, we introduce an orthogonal loss to ensure the independence of different LoRA weights, enabling accurate motion disentanglement.

To flexibly eliminate cinematographic variations caused by ill-posed trajectories, which are difficult to distinguish in LoRA, and to enhance object movement, we also introduce a new camera-free guidance. This technique iteratively executes an extrapolation fusion between different latents during the sampling process of diffusion models, similar to the classifier-free guidance technique (Ho and Salimans 2022).

In brief, our main contributions are as follows:

- ☆ We construct a high-quality video motion dataset with precise trajectory annotations, addressing the lack of such data in the open-source community.
- ☆ We introduce a method to collaboratively optimize LoRA weights in motion ControlNet, effectively separating and controlling camera transitions and object movements
- ☆ We propose camera-free guidance to heuristically eliminate camera transitions caused by multiple trajectories that are challenging to separate with LoRA weights.
- ☆ Extensive experiments demonstrate the superiority of our method in precisely motion control, enabling the generation of videos from images that align with user desires.

Approach

Overview

Image Conductor aims to animate a static image by precisely directing camera transitions and object movements according to user specifications, producing coherent video assets. Our workflow includes trajectory-oriented video data construction, a motion-aware image-to-video architecture, controllable motion separation, and camera-free guidance.

We use user-friendly trajectories to define the intensity and direction of camera transitions and object movements. To address the lack of large-scale annotated video data, we design a data curation pipeline to create a consistent video dataset with appropriate motion.

Using this data, we train video ControlNet (Zhang, Rao, and Agrawala 2023) to synthesize motion-controllable video content. To eliminate ambiguities between camera transitions and object movements, we employ separate sets of LoRA weights. First, we train with camera-only LoRA weights to control camera transitions. Then, we load these

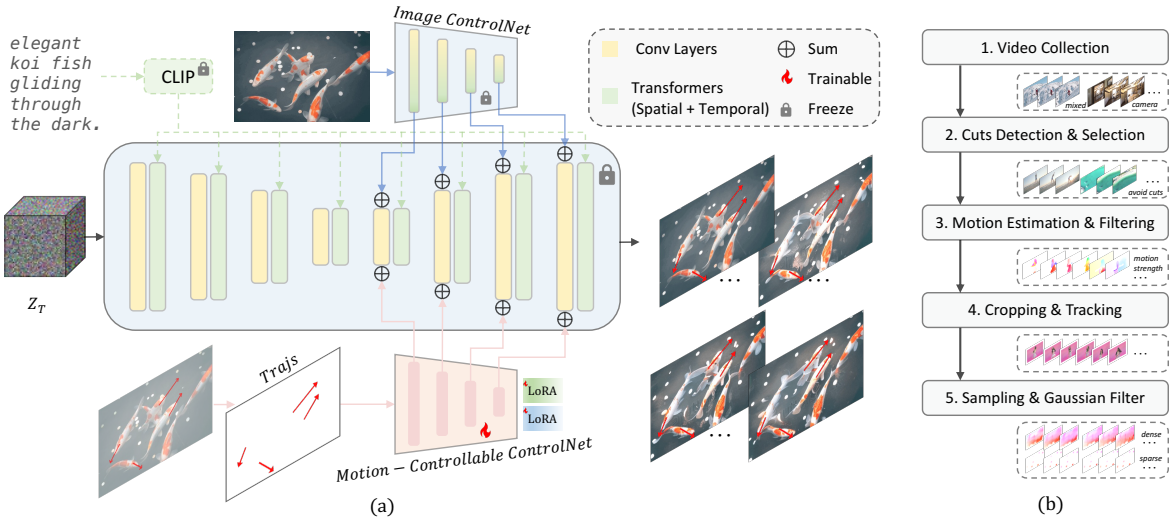

Figure 2: **a) Framework of Image Conductor.** 3D UNet serves as the diffusion backbone, while image ControlNet and motion-controllable ControlNet (and its LoRA weights) convey appearance and motion information, respectively. We progressively fine-tune different modules during trarning phase (see Sec 3.4). **b) Trajectory-oriented video motion data curation workflow.** We carefully curate data to ensure dynamic and consistent video content, as well as precise trajectory annotations.

weights and use a new set of object LoRA weights to decouple object movement, ensuring precise control. We also introduce a loss function with orthogonal constraints to maintain independence between different LoRA weights.

To seamlessly blend camera transitions and object movements, we propose a camera-free guidance technique that iteratively extrapolates between camera and object motion latents during inference. Fig. 2 (a) shows our framework, Fig. 2 (b) illustrates our data curation pipeline, and Fig. 3 presents the core idea of Image Conductor.

Trajectory-Oriented Video Motion Data Curation

Since Image Conductor relies on trajectories to guide motion, we need a dataset with trajectory annotations to track dynamic information in videos. Existing large-scale video datasets typically lack such annotations. While some methods use motion estimators to annotate video data, these approaches often suffer from inaccuracies(Yin et al. 2023; Wang et al. 2023; Wu et al. 2024) or lack generality(Wu et al. 2024). Moreover, almost all annotated datasets with trajectory annotations are not publicly available. To address this, we introduce a comprehensive and general pipeline for generating high-quality video data with appropriate motion and consistent scenes, as illustrated in Fig. 2 (b).

Video Collection. We leverage two datasets in our research: the WebVid dataset (Bain et al. 2021), which is a large-scale mixed dataset with textual descriptions, and the Realestate10K dataset (Zhou et al. 2018), which is a camera-only dataset. The Image Conductor aims to decouple object movements from mixed data, requiring scene consistency and high motion quality. To ensure temporal quality, we process the WebVid dataset by detecting cuts and filtering motion. For the Realestate10K dataset, we focus on the diversity of camera transitions and generate video captions using BLIP2(Li et al. 2023) by extracting frames at specific intervals and concatenating their descriptions.

Cuts Detection and Selection. In videos, cuts refer to transitions between different shots, and generative video models are sensitive to such motion inconsistencies (Blattmann et al. 2023a). To avoid cuts and abrupt scene changes, which can cause the model to overfit these phenomena, we first use a cut detection tool [1] to identify cuts within the video dataset. We then select the longest consistent scenes as our video clips, ensuring scene consistency.

Motion Estimation and Filtering. To ensure the dataset exhibits good dynamics, we use RAFT (Teed and Deng 2020) to compute the optical flow between adjacent frames and calculate the Frobenius norm as a motion score. We filter out the lowest 25% of video samples based on this score. To reduce computational cost, we resize the shorter side of the videos to 256 pixels and randomly sample a 32-frame sequence with a random temporal interval of 1 to 16 frames. These 32 frames are used as the training dataset, and their motion scores are computed for sample filtering.

Cropping and Tracking. To standardize the dimensions of the training data, we perform center cropping on the previously obtained data, resulting in video frames of size $384 \times 256 \times 32$. We then employ CoTracker (Karaev et al. 2023), a tracking method towards dense point, to record motion within the video using a 16×16 grid. Compared to optical flow-based point correspondence methods (Teed and Deng 2020; Xu et al. 2022), tracking avoids drift-induced error accumulation, providing a more accurate representa-

[1] https://github.com/Breakthrough/PySceneDetect.

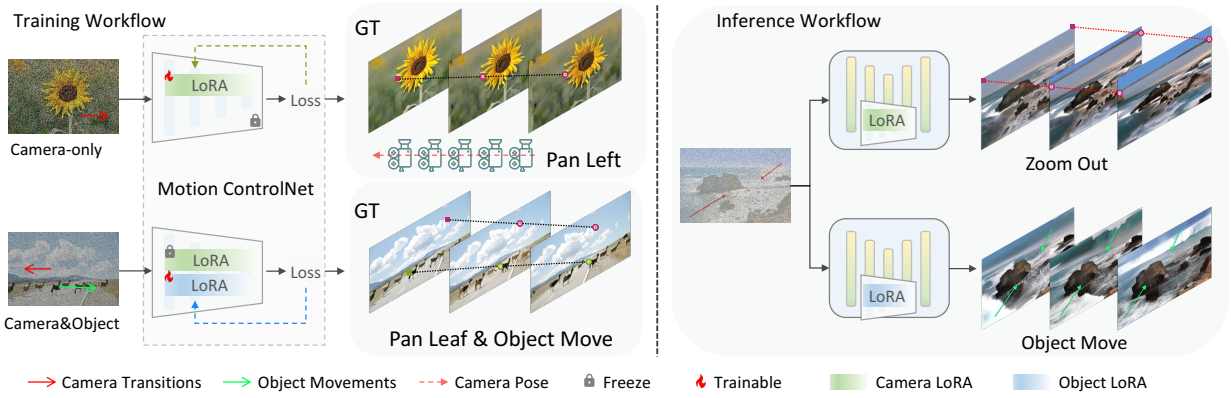

Figure 3: **Fine-grained Motion Separation Method.** a) The training process is divided into two stages. Initially, camera-only data is used to empower the camera LoRA with the ability to control camera transitions. After loading the well-trained camera LoRA, mixed motion data is used to train the object LoRA, refining object motion information. b) During inference, loading different LoRAs provides the model with various control capabilities.

tion of motion. After tracking, we accumulate point trajectories by calculating the differences between adjacent points within the same trajectory. This results in stacked flow maps compatible with the input format of ControlNet (Zhang, Rao, and Agrawala 2023).

Sampling and Gaussian Filter. To enhance user interaction and usability, we use sparse trajectories for motion guidance. We heuristically sample $n \in [1, 8]$ trajectories from the dense set, with 8 being the upper limit. The value of n is randomly selected, and the normalized motion intensity of each trajectory is used as the sampling probability. The accumulated flow map from these trajectories forms a sparse matrix. To avoid training instability caused by the sparse matrix, we apply a Gaussian filter to the trajectories, similar to previous methods (Yin et al. 2023; Wang et al. 2023; Wu et al. 2024). Through this data processing pipeline, we constructed a trajectory-oriented video motion dataset containing 130k mixed videos with camera transitions and object movements, and 62k videos with only camera transitions.

Motion-aware Image-to-video Architecture

Image-to-Video Backbone. As illustrated in Fig. 2 (a), we utilize Animatediff (Guo et al. 2023b) equipped with SparseCtrl (Guo et al. 2023a) as our pre-trained image-to-video foundational model. This model uses the CLIP (Radford et al. 2021) text encoder to extract text embeddings $c_{txt} \in \mathbb{R}^{1\times d}$, which are then passed to the UNet (Ronneberger, Fischer, and Brox 2015) backbone via cross-attention mechanism. The input image, serving as the first frame, is concatenated with an all-zero frame matrix and a mask identifier channel-wise to form $c_{img} \in \mathbb{R}^{T\times 4\times H\times W}$. Next, the video SparseCtrl, a variant of the ControlNet (Zhang and Agrawala 2023) that removes the skip-connections between the ControlNet's and the UNet encoder's input latents, is used to extracts image information from c_{img}. In addition, in the appendix, we also demonstrate our method with another video backbone, namely DynamiCrafter (Xing et al. 2023).

Motion-Controllable ControlNet. To extract motion information from the annotated trajectory input $c_{trajs} \in \mathbb{R}^{T\times 2\times H\times W}$ for composition of camera transitions and object movements in videos, we use ControlNet as the motion encoder to capture multi-level motion representations. This ControlNet incorporates different types of LoRA weights to guide the image-to-video generation with user-desired camera transitions and object movements. Consistent with the observations of SparseCtrl (Guo et al. 2023a), we find that removing the skip connections between the main branch's and the conditional branch's input latents speeds up convergence during training.

Controllable Motion Separation

The aim of our approach is to precisely separate camera transitions and object movements in videos, enabling fine-grained control over the generation of video clip asserts that meets user expectations. To this end, we introduced camera LoRA $\Delta\theta_{\text{cam}}$ and object LoRA $\Delta\theta_{\text{obj}}$ into the motion ControlNet to guide the synthesis of different types of motion. As shown in the Fig. 3, during the training process, we employed a collaborative optimization strategy. First, we optimized the camera LoRA, and then, we optimized the object LoRA based on the loaded camera LoRA. During the inference stage, the model loads different LoRA to control camera transitions (e.g., zooming out) and object movements (e.g., two waves advancing in a specified direction).

Camera Transitions. Since it is available to obtain data with camera-only transition, we straightforwardly train camera LoRA $\theta_{\text{cam}} = \theta_0 + \Delta\theta_{\text{cam}}$ using our our carefully cultivated camera motion dataset, endowing the ControlNet with the ability to direct cinematographic variations. The standard diffusion denoising training objective is utilized:

$$\mathcal{L}_{\text{cam}} = \mathbb{E}_{z_{0,\text{cam}},c,\epsilon\sim\mathcal{N}(0,I),t}\left[\|\epsilon - \epsilon_{\theta_{\text{cam}}}(z_{t,\text{cam}}, t, c)\|_2^2\right], \quad (1)$$

where θ_{cam} is the denoiser with ControlNet's camera LoRA loaded, $z_{t,\text{cam}}$ is the noisy latent of videos with only camera

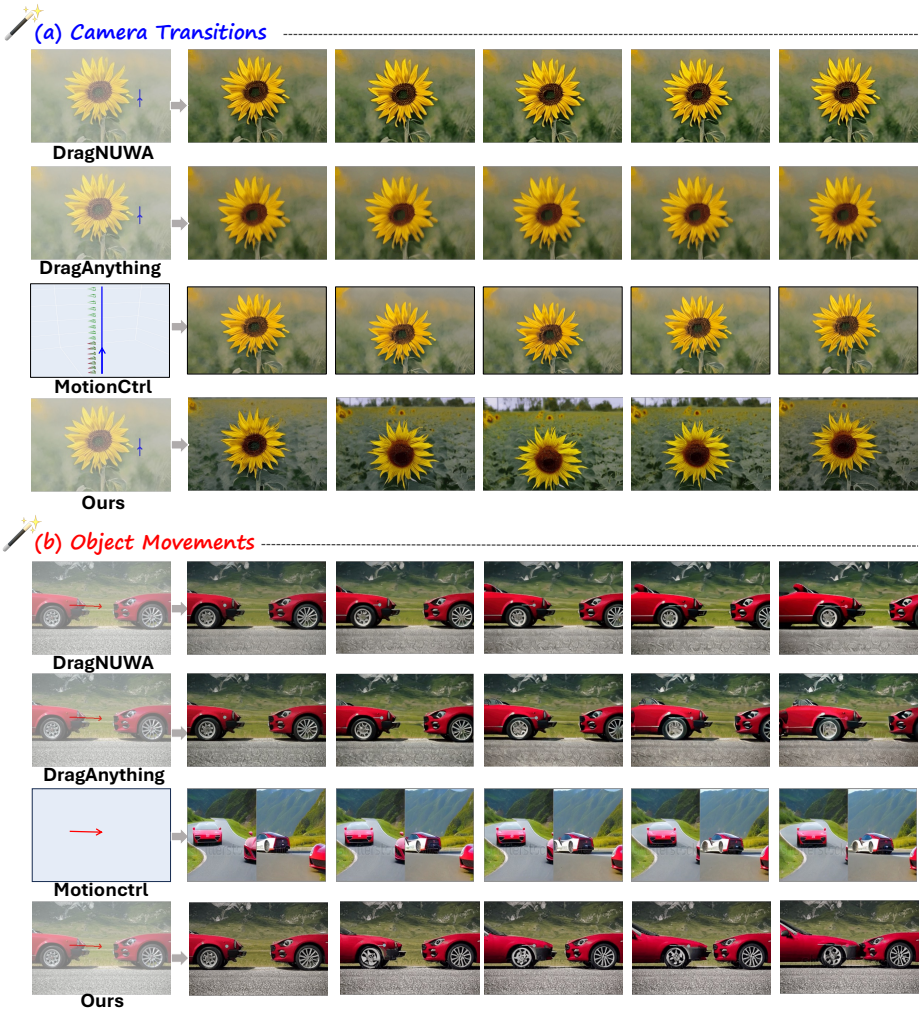

Figure 4: **Qualitative Comparisons of the proposed Image Conductor.** (a) Camera Transitions. Our method can simultaneously utilize text, image, and trajectory prompts as control signals to achieve more natural content and camera transitions. (b) Object Movements. Apart from our method, other approaches incorrectly confuse object movements with camera transitions.

transition at timestep t, while $c = [c_{\text{txt}}, c_{\text{img}}, c_{\text{trajs}}]$ refer to the text prompt, image prompt, and conditional trajectory.

Object Movements. Due to the scarcity of fixed-camera-view video data without cinematographic variations, we need to decouple object motion from mixed data where both camera transitions and object movements are exsist. Observing that distinct types of motion share the same trajectory, we can further train the object LoRA $\theta_{\text{obj}} = \theta_0 + \Delta\theta_{\text{obj}}$ after loading the well-trained camera LoRA weights, i.e., targeting the reconstruction of camera transitions and object movements in the original video content from mixed data. Formally, we load both the camera LoRA and object LoRA simultaneously during training phase, and prevent gradient flow to the camera LoRA via stopgrad $\text{sg}[\cdot]$:

$$\theta_{\text{mix}} = \theta_0 + \text{sg}[\Delta\theta_{\text{cam}}] + \Delta\theta_{\text{obj}}. \tag{2}$$

Similarly, we optimize the object LoRA using the standard diffusion denoising objective:

$$\mathcal{L}_{\text{obj}} = \mathbb{E}_{z_{0,\text{mix}},c,\epsilon\sim\mathcal{N}(0,I),t}\left[\|\epsilon - \epsilon_{\theta_{\text{mix}}}(z_{t,\text{mix}}, t, c)\|_2^2\right], \tag{3}$$

where θ_{mix} is the denoiser with all of ControlNet's LoRA loaded, $z_{t,\text{cam}}$ denotes the noisy latent of videos with camera transition and object movements at timestep t.

Orthogonal Loss. To encourage the object LoRA to learn concepts distinct from the camera LoRA and to accelerate the convergence of the model, we propose an orthogonal loss as a joint optimization objective. Specifically, we extract all linear layer weights W_{cam} and W_{traj} from the different LoRAs and impose an orthogonality constraint on them:

$$\mathcal{L}_{\text{ortho}} = \mathbb{E}_{W_{i,\text{cam}}\in W_{\text{cam}}, W_{i,\text{traj}}\in W_{\text{traj}}}\left[\|I - W_{i,\text{cam}} W_{i,\text{traj}}^T\|_2^2\right] \tag{4}$$

where I represents the identity matrix, $W_{i,\text{cam}}$ and $W_{i,\text{traj}}$ refer to the weights of the i-th linear layer of the camera LoRA and object LoRA, respectively.

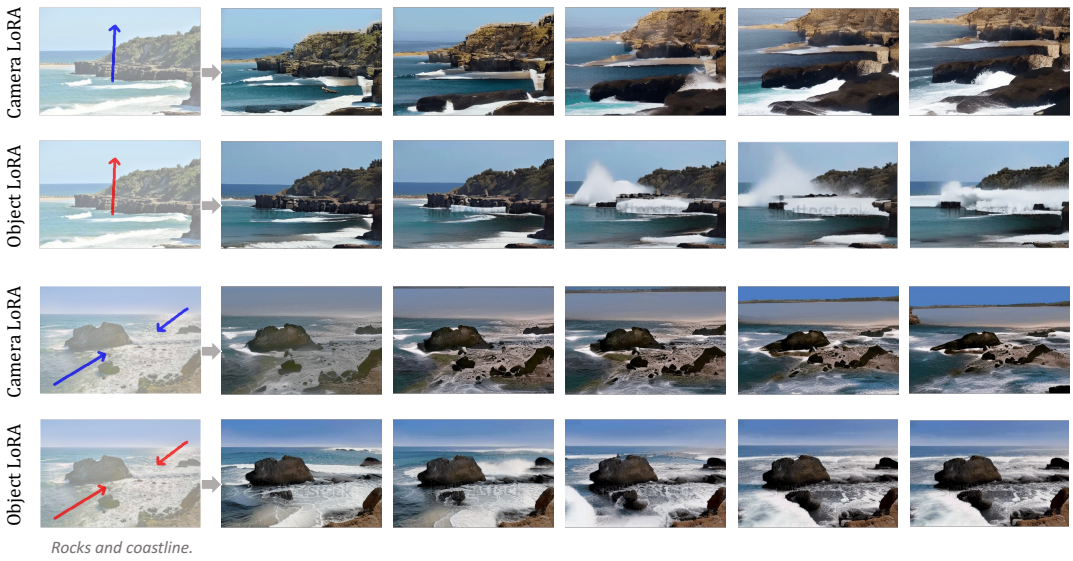

Figure 5: **Effect of distinct LoRA weights.** Image conductor enables users to independently control distinct motion.

Method	Automatic Metrics				Human Evaluation		
	FID ↓	FVD ↓	CamMC ↓	ObjMC ↓	Overall ↑	Quality ↑	Motion ↑
DN (Yin et al. 2023)	237.26	1283.85	48.72	51.24	31.8%	37.1%	27.7%
DA (Wu et al. 2024)	243.17	1287.15	66.54	60.97	6.5%	8.1%	6.3%
Image Conductor	**209.74**	**1116.17**	**33.49**	**42.38**	**61.7%**	**54.8%**	**66.0%**

Table 1: **Quantitative Comparisons with SOTA Methods.** We utilize automatic metrics (*i.e.*, FID, FVD, CamMC, ObjMC) and human evaluation (*i.e.*, overall performance, sample quality, motion similarity) to evaluate the performance. DN and DA denotes DragNUWA (Wu et al. 2024) and DragAnything (Yin et al. 2023), respectively.

In all, the optimization process is incremental. We first optimize the camera LoRA using $\mathcal{L}_{\text{cam}}$, and then optimize the object LoRA using $\mathcal{L}_{\text{mix}}$ and $\mathcal{L}_{\text{ortho}}$.

Camera-free Guidance

Multiple object control often leads to motion ambiguity. Inspired by classifier-free guidance (Ho and Salimans 2022), we propose a camera-free guidance technique to flexibly and seamlessly enhance motion intensity while eliminating camera transitions:

$$\begin{aligned}\hat{\epsilon}_{\theta_0,\theta_{\text{trajs}}}(\boldsymbol{x}_t,\boldsymbol{c}) =& \epsilon_{\theta_0}(\boldsymbol{x}_t,\varnothing) \\ &+ \lambda_{\text{cfg}}(\epsilon_{\theta_0}(\boldsymbol{x}_t,\boldsymbol{c}) - \epsilon_{\theta_0}(\boldsymbol{x}_t,\varnothing)) \\ &+ \lambda_{\text{trajs}}(\epsilon_{\theta_{\text{trajs}}}(\boldsymbol{x}_t,\boldsymbol{c}) - \epsilon_{\theta_0}(\boldsymbol{x}_t,\boldsymbol{c})),\end{aligned} \tag{5}$$

where θ_{trajs} refers to the model with object LoRA and θ_0 is the model without any LoRA. The final output latent is derived by extrapolating the outputs of these two components.

Experiments

Comparisons with State-of-the-Art Methods

We compare Image Conductor with SOTA image-based or text-based motion controllable video generation methods, namely DragNUWA (Yin et al. 2023), DragAnything (Wu et al. 2024) and MotionCtrl (Wang et al. 2023).

Evaluation datasets. To independently evaluate camera transitions and object movements, we use two distinct datasets: 1) Camera-only Motion Evaluation Dataset: we select 10 camera trajectories, *e.g.* pan left, pan right, pan up, pan down, zoom in, zoom out, to evaluate control over cinematographic variations. 2) Object-only Motion Evaluation Dataset: we design 10 varied trajectories, including straight lines, curves, shaking lines, and their combinations.

Evaluation metrics. To thoroughly evaluate the effectiveness of our method, we following MotionCtrl (Wang et al. 2023) to assessed two types of metrics: 1) Video content quality evaluation. We employ Fréchet Inception Distance (FID)(Heusel et al. 2017), Fréchet Video Distance (FVD)(Unterthiner et al. 2018) to measure the visual quality and temporal coherenceand. The reference videos of FID and FVD are 5000 videos randomly selected from WebVid (Bain et al. 2021). 2) Video motion quality evaluation. The Euclidean distance between the predicted and ground truth trajectories, i.e., CamMC and ObjMC, is used to evaluate the motion control.

Implementation details. We use Animatediff v3 (Guo et al. 2023b) as our base model for image-to-video generation. We train only the motion ControlNet while keeping the UNet backbone weights frozen. Details are in the appendix.

Figure 6: **Effect of Camera-free Guidance.** The camera-free guidance flexibly enhances object movements during inference.

Qualitative Evaluation. Fig. 4 displays some of our qualitative results. Compared to previous methods (Yin et al. 2023; Wu et al. 2024; Wang et al. 2023), our approach can effectively control camera transitions and object movements. In terms of camera transitions, both DragNUWA and DragAnything fail to achieve the camera transition of panning down and then up in the generated video. Although Motionctrl-SVD is capable of generating the specified camera movement, it is unable to define natural content changes via text prompts. Additionally, it cannot accurately define the intensity of camera changes, and sometimes introduces distortion artifact.

In terms of object movements, both DragNUWA and DragAnything incorrectly interpret object movement as camera transition, resulting in generated videos that do not meet user intentions. In addition, the motion trajectories of their generated videos are often poorly matched to the desired trajectories precisely due to the errors introduced by the labeled dataset. As trajectory-based MotionCtrl relies on the text-to-video model, we directly use text and trajectory prompts to control the generation of the video under different seed. The results demonstrate that it lacks fine-grained control over the generated content due to its inability to use images as conditions. Additionally, it still exhibits a significant amount of camera transition rather than object movement. In all, our method is capable of accurately and finely controlling various types of motion utilizing the separated LoRA.

Quantitative Evaluation. As shown in the Tab. 1, compared to other methods, our proposed Image Conductor achieves state-of-the-art quantitative performance. We measure our alignment with the given trajectoies via the CamMC and ObjMC metrics, surpassing the baseline models and demonstrating our precise motion control capabilities. At the same time, the FID and FVD metrics illustrate that our generation quality surpasses other models, capable of producing realistic videos. Furthermore, we invite 31 participants to assess the results of DragNUWA, DragAnything and Image Conductor. The assessment includes video quality, motion similarity. Participants are also asked to give an overall preference for each compared pair. The statistical results confirm that our generated videos not only appear more realistic and visually appealing but also exhibit superior motion adherence compared to those produced by other models.

Ablation Studies

Effect of Distinct LoRA Weights. To validate our interactive optimization strategy, which uses distinct LoRA weights to separate camera transitions from object movements, we guide different LoRA models with the same trajectory to generate videos. As shown in Fig. 5, loading various LoRA weights endows the model with different capabilities. For instance, a vertically upward trajectory causes the video to pan up when using the camera LoRA, and it generates upward waves when the object LoRA is applied.

Effect of Camra-free Guidance. As shown in Fig. 6, using camera-free guidance can facilitate the separation of object movements from camera transitions in several challenging examples. When camera-free guidance λ_{trajs} is set to 1, *i.e.*, camera-free guidance is not yet used, the generated video exhibits a unexpected pan left transformation. When the λ_{trajs} is set to 1.1, the generated videos exhibit reasonable object movements, yet some artifacts still remain. As the guidance increases, the movements of the object becomes more apparent and clear.

Conclusion

In conclusion, this paper introduces Image Conductor, a novel approach for precise and fine-grained control of camera transitions and object movements in interactive video synthesis. We design a training strategy and utilized distinct LoRA weights to decouple camera transition and object movements. Additionally, we propose a camera-free guidance technique to enhance object movement control. Extensive experiments demonstrate the effectiveness of our method, marking a significant step towards practical applications in video-centric creative expression.

Acknowledgments

This work was was partially supported by NSFC (No. 62176008) and Guangdong Provincial Key Laboratory of Ultra High Definition Immersive Media Technology(Grant No. 2024B1212010006).

References

Bain, M.; Nagrani, A.; Varol, G.; and Zisserman, A. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1728–1738.

Bar-Tal, O.; Chefer, H.; Tov, O.; Herrmann, C.; Paiss, R.; Zada, S.; Ephrat, A.; Hur, J.; Li, Y.; Michaeli, T.; et al. 2024. Lumiere: A space-time diffusion model for video generation. *arXiv preprint arXiv:2401.12945*.

Blattmann, A.; Dockhorn, T.; Kulal, S.; Mendelevitch, D.; Kilian, M.; Lorenz, D.; Levi, Y.; English, Z.; Voleti, V.; Letts, A.; et al. 2023a. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*.

Blattmann, A.; Rombach, R.; Ling, H.; Dockhorn, T.; Kim, S. W.; Fidler, S.; and Kreis, K. 2023b. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22563–22575.

Brooks, T.; Peebles, B.; Holmes, C.; DePue, W.; Guo, Y.; Jing, L.; Schnurr, D.; Taylor, J.; Luhman, T.; Luhman, E.; Ng, C.; Wang, R.; and Ramesh, A. 2024. Video generation models as world simulators.

Chen, H.; Xia, M.; He, Y.; Zhang, Y.; Cun, X.; Yang, S.; Xing, J.; Liu, Y.; Chen, Q.; Wang, X.; et al. 2023. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*.

Girdhar, R.; Singh, M.; Brown, A.; Duval, Q.; Azadi, S.; Rambhatla, S. S.; Shah, A.; Yin, X.; Parikh, D.; and Misra, I. 2023. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*.

Guo, Y.; Yang, C.; Rao, A.; Agrawala, M.; Lin, D.; and Dai, B. 2023a. Sparsectrl: Adding sparse controls to text-to-video diffusion models. *arXiv preprint arXiv:2311.16933*.

Guo, Y.; Yang, C.; Rao, A.; Wang, Y.; Qiao, Y.; Lin, D.; and Dai, B. 2023b. AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning. *arXiv preprint arXiv:2307.04725*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

Ho, J.; Chan, W.; Saharia, C.; Whang, J.; Gao, R.; Gritsenko, A.; Kingma, D. P.; Poole, B.; Norouzi, M.; Fleet, D. J.; et al. 2022. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*.

Ho, J.; and Salimans, T. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Karaev, N.; Rocco, I.; Graham, B.; Neverova, N.; Vedaldi, A.; and Rupprecht, C. 2023. Cotracker: It is better to track together. *arXiv preprint arXiv:2307.07635*.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*.

Teed, Z.; and Deng, J. 2020. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, 402–419. Springer.

Unterthiner, T.; Van Steenkiste, S.; Kurach, K.; Marinier, R.; Michalski, M.; and Gelly, S. 2018. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*.

Wang, J.; Zhang, Y.; Zou, J.; Zeng, Y.; Wei, G.; Yuan, L.; and Li, H. 2024. Boximator: Generating Rich and Controllable Motions for Video Synthesis. *arXiv preprint arXiv:2402.01566*.

Wang, Z.; Yuan, Z.; Wang, X.; Chen, T.; Xia, M.; Luo, P.; and Shan, Y. 2023. Motionctrl: A unified and flexible motion controller for video generation. *arXiv preprint arXiv:2312.03641*.

Wu, W.; Li, Z.; Gu, Y.; Zhao, R.; He, Y.; Zhang, D. J.; Shou, M. Z.; Li, Y.; Gao, T.; and Zhang, D. 2024. DragAnything: Motion Control for Anything using Entity Representation. *arXiv preprint arXiv:2403.07420*.

Xing, J.; Xia, M.; Zhang, Y.; Chen, H.; Wang, X.; Wong, T.-T.; and Shan, Y. 2023. Dynamicrafter: Animating open-domain images with video diffusion priors. *arXiv preprint arXiv:2310.12190*.

Xu, H.; Zhang, J.; Cai, J.; Rezatofighi, H.; and Tao, D. 2022. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8121–8130.

Yin, S.; Wu, C.; Liang, J.; Shi, J.; Li, H.; Ming, G.; and Duan, N. 2023. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*.

Zhang, L.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3836–3847.

Zhao, W.; Liu, S.; Guo, H.; Wang, W.; and Liu, Y.-J. 2022. Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In *ECCV*.

Zhou, T.; Tucker, R.; Flynn, J.; Fyffe, G.; and Snavely, N. 2018. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*.